

This Page Is Inserted by IFW Operations  
and is not a part of the Official Record

## **BEST AVAILABLE IMAGES**

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images may include (but are not limited to):

- BLACK BORDERS
- TEXT CUT OFF AT TOP, BOTTOM OR SIDES
- FADED TEXT
- ILLEGIBLE TEXT
- SKEWED/SLANTED IMAGES
- COLORED PHOTOS
- BLACK OR VERY BLACK AND WHITE DARK PHOTOS
- GRAY SCALE DOCUMENTS

**IMAGES ARE BEST AVAILABLE COPY.**

**As rescanning documents *will not* correct images,  
please do not report the images to the  
Image Problem Mailbox.**

Application No.09/874,167

**AMENDMENTS TO THE CLAIMS:**

This listing of claims will replace all prior versions and listings of claims in the application:

**LISTING OF CLAIMS:**

1. (Currently Amended) A problem solving method for use in a computer system, wherein the computer system includes an applications module having a problem solver, wherein the problem solver comprises a solving module and a complexity module, the method comprising ~~the steps of:~~
  - receiving a problem statement from the applications module;
  - configuring the solving module with configuration parameters;
  - determining an expected solver behavior associated with said configuration parameters for said problem statement;
  - determining actual solver behavior;
  - reviewing said actual solver behavior to determine if a problem solution has been found;
  - determining whether to perform a solver iteration step or to request the complexity module to perform an adaptation step if a problem solution has not been found, wherein said complexity module includes means for detecting non-input-output mapping optimization problems, means for estimating rate of improvement bounds, and means for selecting a complexity related action, wherein said complexity related action comprises selecting at least one member from the group consisting of returning an error, accepting the best result possible for fixed computational resources, adding computational resources, and changing the problem statement, wherein the complexity module captures previous data on said expected solver behavior as a function of said problem configuration parameters ~~and performs at least one of performing a solver iteration step, altering said problem definition, changing the optimization algorithm, or requesting additional system resources based on the complexity module's knowledge of optimization behavior.~~

Application No.09/874,167

performing a said solver iteration step when said solver iteration step is selected, comprising the steps of determining new actual solver behavior and determining whether to repeat said solver iteration step;

repeating said solver iteration step until said adaptation step is selected;

comparing said actual solver behavior with said expected solver behavior when said adaptation step is selected;

requesting the complexity module to perform said adaptation step;

performing said adaptation step, comprising the steps of modifying said configuration parameters within the complexity module, configuring the solving module with said modified configuration parameters, determining expected solver behavior associated with said modified configuration parameters for said problem statement, selecting an algorithm to calculate a revised problem solution, determining a revised actual solver behavior associated with said modified configuration parameters for said problem statement, reviewing said revised actual solver behavior to determine if a problem solution has been found, determining whether to perform said solver iteration step or to request the complexity module to perform another adaptation step if a problem solution has not been found, and repeating said iteration step until said adaptation step is selected;

repeating said adaptation step until a problem solution is found; and providing the solution to the applications module.

2. (Original) The problem solving method according to claim 1, further comprising the step of selecting an algorithm to calculate an initial problem solution.

3. (Original) The problem solving method according to claim 1, further comprising the step of refining said configuration parameters.

4. (Original) The problem solving method according to claim 1, wherein the problem solver comprises an adaptive constraint problem solver.

Application No.09/874,167

5. (Original) The problem solving method according to claim 1, further comprising the step of transforming said problem statement after receiving said problem statement from the applications module.

6. (Original) The problem solving method according to claim 5, further comprising the step of transforming said problem solution before providing said problem solution to the applications module.

7. (Original) The problem solving method according to claim 1, wherein said configuration parameters include problem configuration parameters and solver configuration parameters.

8. (Original) The problem solving method according to claim 7, further comprising the step of transforming said problem configuration parameters before providing said problem configuration parameters to the solving module.

9. (Currently Amended) A computer system for problem solving, the system having implementation units communicating with the computer system, the system comprising:

an input device for providing the problem statement;

a computer coupled to the output of said input device includes an applications module having a problem solver, wherein the problem solver comprises a solving module and a complexity module;

a memory portion coupled to said computer, wherein said memory portion includes an applications module having a problem solver, wherein the problem solver includes a solving module and a complexity module, and wherein said complexity module includes means for detecting non-input-output mapping optimization problems, means for estimating rate of improvement bounds, and means for selecting a complexity related action, wherein said complexity related action comprises selecting at least one member from the group consisting of returning an error, accepting the best result possible

Application No.09/874,167

for fixed computational resources, adding computational resources, and changing the problem statement, said memory portion comprising:

- software for receiving the problem statement from said input device;
- software for identifying system configuration parameters and system secondary goals;
- software for configuring a problem solver;
- software for determining expected solver behavior;
- software for determining actual solver behavior and determining whether a solution has been found;
- software for determining whether to perform a solver iteration step or to perform an adaptation step, comprising capturing previous data on said expected solver behavior as a function of said problem configuration parameters and performing at least one of performing said solver iteration step, altering said problem definition, changing the optimization algorithm, or requesting additional system resources based on knowledge of optimization behavior; and
- software for performing an adaptation step, comprising modifying said configuration parameters and reconfiguring said problem solver; and
- output means for providing a solution statement.

10. (Original) The computer system for problem solving according to claim 9, wherein said problem solver comprises an adaptive constraint problem solver.

11. (Original) The computer system for problem solving according to claim 9, wherein said memory portion further comprises software including a learning module for refining said expected solver behavior.

12. (Original) The computer system for problem solving according to claim 9, further comprising a problem transformer module for transforming said problem statement after receiving said problem statement from said input device.

Application No.09/874,167

13. (Original) The computer system for problem solving according to claim 9, further comprising a problem transformer module for transforming said problem solution before providing said problem solution to said output device.

14. (Original) The computer system for problem solving according to claim 9, wherein said configuration parameters include problem configuration parameters and solver configuration parameters.

15. (Original) The computer system for problem solving according to claim 14, further comprising a problem transformer module for transforming said problem configuration parameters before providing said problem configuration parameters to said solving module.

16. (Original) The computer system for problem solving according to claim 9, wherein said software for determining expected solver behavior comprises a data structure, said data structure containing configuration parameters and expected behaviors for a plurality of problem types.

17. (Original) The computer system for problem solving according to claim 9, wherein said control computer comprises an embedded computer.

18. (Original) The computer system for problem solving according to claim 17, wherein said embedded computer system controls at least one operation within a copier or printer.

19. (Original) The computer system for problem solving according to claim 17, wherein said embedded computer system controls at least one operation within a process control system.

Application No.09/874,167

20. (Original) The computer system for problem solving according to claim 17, wherein said embedded computer system controls at least one operation within a diagnostics unit.

21. (Currently Amended) A computer system for problem solving, the system having implementation units communicating with the computer system, the system comprising:

- an input device for providing the primary goal for the task to be performed;
- a computer coupled to the output of said input device;
- a memory portion coupled to said computer comprising:
- a controllable solving module for calculating actual solver behavior;
- a complexity module coupled to said controllable solving module, for configuring a problem statement, wherein said complexity module includes means for detecting non-input-output mapping optimization problems, means for estimating rate of improvement bounds, and means for selecting a complexity related action, wherein said complexity related action comprises selecting at least one member from the group consisting of returning an error, accepting the best result possible for fixed computational resources, adding computational resources, and changing the problem statement, and wherein the complexity module captures previous data on said expected solver behavior as a function of said problem configuration parameters and performs at least one of performing a solver iteration step, altering said problem definition, changing the optimization algorithm, or requesting additional system resources based on the complexity module's knowledge of optimization behavior; and
- a comparison module for comparing said actual solver behavior with expected solver behavior; and
- output means for providing a solution statement.

22. (Original) The computer system for problem solving according to claim 21, wherein said problem solver comprises an adaptive constraint problem solver.

Application No.09/874,167

23. (Original) The computer system for problem solving according to claim 21, further comprising a learning module for refining said expected solver behavior.

24. (Original) The computer system for problem solving according to claim 21, further comprising a problem transformer module for transforming said problem statement after receiving said problem statement from said input device.

25. (Original) The computer system for problem solving according to claim 21, further comprising a problem transformer module for transforming said problem solution before providing said problem solution to said output means.

26. (Original) The computer system for problem solving according to claim 21, wherein said configuration parameters include problem configuration parameters and solver configuration parameters.

27. (Original) The computer system for problem solving according to claim 26, further comprising a problem transformer module for transforming said problem configuration parameters before providing said problem configuration parameters to said solving module.

28. (Original) The computer system for problem solving according to claim 21, wherein said complexity module a data structure, said data structure containing configuration parameters and expected behaviors for a plurality of problem types.

29. (Original) The computer system for problem solving according to claim 21, wherein said control computer comprises an embedded computer system.



Application No.09/874,167

30. (Original) The computer system for problem solving according to claim 29, wherein said embedded computer system controls at least one operation within a copier or printer.

31. (Original) The computer system for problem solving according to claim 29, wherein said embedded computer system controls at least one operation within a process control system.

32. (Original) The computer system for problem solving according to claim 29, wherein said embedded computer system controls at least one operation within a diagnostics unit.

33. (Currently Amended) A problem solver within a computer system, said problem solver comprising:

means for receiving a problem statement;

means for determining expected solver behavior associated with said problem statement;

means for providing configuration parameters for a plurality of problems;

means for calculating actual solver behavior;

means for reviewing said actual solver behavior to determine if a problem solution has been found;

means for determining whether to perform a solver iteration step or to request an adaptation step if a problem solution has not been found, comprising a complexity module, wherein said complexity module includes means for detecting non-input-output mapping optimization problems, means for estimating rate of improvement bounds, and means for selecting a complexity related action, wherein said complexity related action comprises selecting at least one member from the group consisting of returning an error, accepting the best result possible for fixed computational resources, adding computational resources, and changing the problem statement, and wherein said complexity module captures previous data on said expected solver behavior as a function

Application No.09/874,167

~~of said problem configuration parameters and performs at least one of performing said solver iteration step, altering said problem definition, changing the optimization algorithm, or requesting additional system resources based on the complexity module's knowledge of optimization behavior;~~

means for performing a solver iteration step, comprising performing another search step, calculating a revised actual solver behavior and determining whether to repeat said solver iteration step;

means for comparing said actual solver behavior with said expected solver behavior;

means for requesting performance of an adaptation step;

means for performing an adaptation step, comprising modifying said configuration parameters, determining a revised expected solver behavior, and providing said modified configuration parameters and said revised expected solver behavior to said means for performing a solver iteration step; and

means for providing the problem solution to an output device.

34. (Currently Amended) A problem solving method for use in a computer system comprising the steps of:

receiving a problem statement;

configuring the problem solver with configuration parameters;

determining expected solver behavior associated with said configuration parameters for said problem statement;

determining actual solver behavior;

determining if a problem solution has been found;

determining, within a complexity module, whether to perform a solving iteration step or an adaptation step if a problem solution has not been found, wherein said complexity module includes means for detecting non-input-output mapping optimization problems, means for estimating rate of improvement bounds, and means for selecting a complexity related action, wherein said complexity related action comprises selecting at least one member from the group consisting of returning an error, accepting the best

Application No.09/874,167

result possible for fixed computational resources, adding computational resources, and changing the problem statement, and wherein said complexity module captures previous data on said expected solver behavior as a function of said problem configuration parameters and performs at least one of performing said solver iteration step, altering said problem definition, changing the optimization algorithm, or requesting additional system resources based on the complexity module's knowledge of optimization behavior;

performing said solver iteration step, when said solver iteration step is selected, comprising the steps of determining a new actual solver behavior and determining whether to repeat said iteration step;

repeating said solver iteration step until said adaptation step is selected;

comparing said actual solver behavior with said expected solver behavior when said adaptation step is selected;

performing said adaptation step, comprising the steps of modifying said configuration parameters, determining expected solver behavior associated with said modified configuration parameters, determining a revised actual solver behavior, reviewing said revised actual solver behavior to determine if a problem solution has been found, determining whether to perform said solver iteration step or to perform another adaptation step if a problem solution has not been found, and repeating said iteration step until said adaptation step is selected;

repeating said adaptation step until a problem solution is found; and  
transmitting a solution statement.

35. (Original) The problem solving method according to claim 34, wherein said problem solving method comprises an adaptive constraint problem solving method.

36. (Original) The problem solving method according to claim 34, further comprising the step of refining the control parameters.

Application No.09/874,167

37. (Original) The problem solving method according to claim 34 further comprising the step of transforming said problem statement.

38. (Original) The problem solving method according to claim 34, further comprising the step of transforming said problem solution.

39. (Original) The problem solving method according to claim 34, wherein said configuration parameters include problem configuration parameters and solver configuration parameters.

40. (Original) The problem solving method according to claim 39, further comprising the step of transforming said problem configuration parameters.

41. (Original) The problem solving method according to claim 34, further comprising the step of selecting an algorithm for calculating a problem solution.

42. (Canceled).

43. (New) An article of manufacture comprising a computer usable medium having computer readable program code embodied in said medium which, when said program code is executed by said computer causes said computer to perform method steps for problem solving for use in a computer system, wherein the problem solver includes a solving module and a complexity module, said method comprising:

- receiving a problem statement from the applications module;
- configuring the solving module with configuration parameters;
- determining an expected solver behavior associated with said configuration parameters for said problem statement;
- determining actual solver behavior;
- reviewing said actual solver behavior to determine if a problem solution has been found;

Application No.09/874,167

determining whether to perform a solver iteration step or to request the complexity module to perform an adaptation step if a problem solution has not been found, wherein said complexity module includes means for detecting non-input-output mapping optimization problems, means for estimating rate of improvement bounds, and means for selecting a complexity related action, wherein said complexity related action comprises selecting at least one member from the group consisting of returning an error, accepting the best result possible for fixed computational resources, adding computational resources, and changing the problem statement, and wherein the complexity module captures previous data on said expected solver behavior as a function of said problem configuration parameters;

performing a said solver iteration step when said solver iteration step is selected, comprising the steps of determining new actual solver behavior and determining whether to repeat said solver iteration step;

repeating said solver iteration step until said adaptation step is selected;

comparing said actual solver behavior with said expected solver behavior when said adaptation step is selected;

requesting the complexity module to perform said adaptation step;

performing said adaptation step, comprising the steps of modifying said configuration parameters within the complexity module, configuring the solving module with said modified configuration parameters, determining expected solver behavior associated with said modified configuration parameters for said problem statement, selecting an algorithm to calculate a revised problem solution, determining a revised actual solver behavior associated with said modified configuration parameters for said problem statement, reviewing said revised actual solver behavior to determine if a problem solution has been found, determining whether to perform said solver iteration step or to request the complexity module to perform another adaptation step if a problem solution has not been found, and repeating said iteration step until said adaptation step is selected;

repeating said adaptation step until a problem solution is found; and

providing the solution to the applications module.